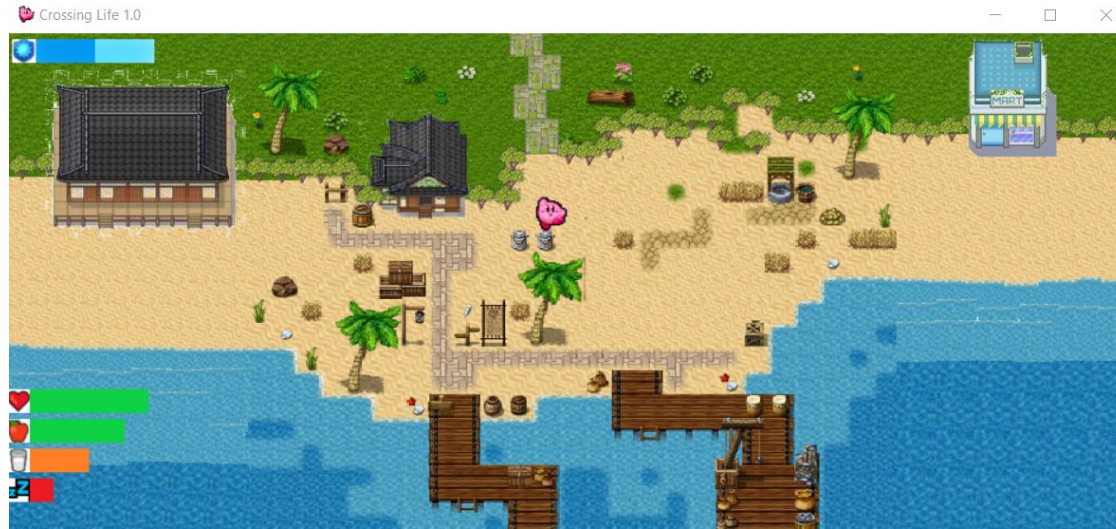


# Life Crossing

Simule le concept de vie artificielle, ce jeu est inspiré d'Animal crossing.







# Personnage.h

Ce module permet de gérer les différentes caractéristiques du personnes.

*Données membre :*

**String nom** : nom du personnage

**char avatar** : l'avatar du personnage

**int argent** : l'argent du personnage

**Point2D position** : la position du personnage

**Vie vie** : l'argent du personnage

**Jauge niveau** : le niveau

**Jauge xp** : les points d'xp du personnage

**Inventaire inventaire** : l'inventaire du personnage

*Fonctions membres :*

**Personnage()** : crée un personnage avec des données initiales

**Mutateurs/seteurs** :sert à accéder et modifier les données passées en privé

**bas(Terrain &t), haut(Terrain &t), droite(Terrain &t), gauche(Terrain &t)**:sert à modifier la position du personnage dans le terrain passé en argument

**void affichePersonnage()** : affiche les données du personnage

**void varieAuto()** : effectue les changements des données de niveaux et d'xp en fonction de l'avancement de la partie

**void testRegression()** : test toutes le fonctions du module avec des assert.

# sdlJeu.h

Ce module fait le lien entre tous les modules implémentés et les images du jeux.

**void jouer(SDL\_Surface \* ecran,Jeu & jeu,Map & map) :** gère toutes les actions du terrain principal, coeur du jeu version graphique

**void teleporter(SDL\_Surface \* ecran,int nb\_carte,Jeu & jeu,Map & map) :** gère le changement de carte et actions réalisées en dehors du terrain principal

**void deplacerJoueur(SDL\_Rect \*pos,int direction,SDL\_Surface \*ecran,int & n,int nb\_carte,Jeu & jeu,Map & map) :** gère les déplacements du joueur et les collisions

**void sdlPlanter(SDL\_Surface \*ecran,const SDL\_Rect \*pos,bool &b,Jeu & jeu,Map & map) :** gère l'interface jardin

**string transformeConstantes(int n) :** effectue la conversion d'un int en string

**void dialoguePNJ(SDL\_Surface \* ecran,SDL\_Rect \*pos,Map & map) :** gère les dialogues avec les PNJ

**void affichageJauge(SDL\_Surface \* ecran, Jeu & jeu) :** affiche les jauges

**void affichageMission(SDL\_Surface \* ecran, Jeu & jeu) :** affiche la mission en cours

**void affichageReussie(SDL\_Surface \* ecran, Jeu & jeu, int miss) :** affiche un message en cas de succès d'une mission

**void manger(Jeu & jeu) :** fait manger le personnage

**void dormir(Jeu & jeu) :** fait dormir le personnage

**void boire(Jeu & jeu) :** fait boire le personnage

# Conclusion

**Réalisations** : Déplacement, vie (dormir, manger et boire), remporter et perdre de l'argent, planter, récolter, dialoguer, effectuer des missions, acheter de l'eau et des graines, et vendre nos fruits et légumes.

**Non réalisé** : Affichage de texte sur la fenêtre (pour afficher l'argent ou le niveau du personnage par exemple), on a cependant contré cela pour les dialogues en affichant des images de dialogue. Système d'avatar pour changer l'apparence. Création du jardin dans la version texte ainsi que la vente des fruits légumes, pas le temps en version texte.

**Avec plus de temps** : Nouveaux fruits et légumes disponibles selon le niveau par exemple. Implémentation de plus d'activités, missions, ... création de minis jeux interactifs en guise de missions(défi) et sauvegarde de la partie. Système de sauvegarde. Davantage de missions

**Difficultés** : Gestion de la variation des jauges (pas lié au temps mais aux nombres de pas). Pas de temps de pousse pour les fruits/légumes. En générale, la création d'une fonction qui dépend du temps. Gestion de "level up" lorsque la jauge d'xp est pleine

**Respect du cahier des charges** : Un peu de retard sur la version txt, + de temps que prévu. Délais d'implémentation des modules et version graphique respectés. Utilisation de la lib SDL1 au lieu de SDL2.